



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/847,781	05/01/2001	Martin Matula	SUN-P5775	6697
7590	05/19/2004		EXAMINER	
David B. Ritchie Thelen Reid & Priest LLP P. O. Box 640640 San Jose, CA 95164-0640			CAO, DIEM K	
			ART UNIT	PAPER NUMBER
			2126	
			DATE MAILED: 05/19/2004	

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/847,781	MATULA ET AL.
Examiner	Art Unit	
Diem K Cao	2126	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 01 May 2001.

2a) This action is **FINAL**. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-66 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-66 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.

 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) Notice of References Cited (PTO-892)
2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 2.
4) Interview Summary (PTO-413)
Paper No(s)/Mail Date. .
5) Notice of Informal Patent Application (PTO-152)
6) Other: .

DETAILED ACTION

1. This Office action is in response to the application filed on May 1st 2001.
2. Claims 1-66 are presented for examination.

Double Patenting

3. A rejection based on double patenting of the "same invention" type finds its support in the language of 35 U.S.C. 101 which states that "whoever invents or discovers any new and useful process ... may obtain a patent therefor ..." (Emphasis added). Thus, the term "same invention," in this context, means an invention drawn to identical subject matter. See *Miller v. Eagle Mfg. Co.*, 151 U.S. 186 (1894); *In re Ockert*, 245 F.2d 467, 114 USPQ 330 (CCPA 1957); and *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970).

A statutory type (35 U.S.C. 101) double patenting rejection can be overcome by canceling or amending the conflicting claims so they are no longer coextensive in scope. The filing of a terminal disclaimer cannot overcome a double patenting rejection based upon 35 U.S.C. 101.

4. Claim 1 is provisionally rejected under 35 U.S.C. 101 as claiming the same invention as that of claim 1 of copending Application No. 09/848,392. This is a provisional double patenting rejection since the conflicting claims have not in fact been patented.

5. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

6. Claims 1, 17, 34 and 50 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 67, 83, 100 and 111 of copending Application No. 09/848,392. Although the conflicting claims are not identical, they are not patentably distinct from each other because the only difference between the instant application and the copending application is the instant application claims a method for dynamic implementation of Java Metadata Interface to a metamodel while the copending application claims an apparatus for dynamic implementation of Java Metadata Interface to a metamodel.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Allowable Subject Matter

7. Claims 3-4, 6-8, 10-16, 20-21, 23-25, 27-33, 36-37, 39-41, 43-49, 53-54, 56-58, and 60-66 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

Art Unit: 2126

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. Claims 1, 17-18, 34, and 50-51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Admitted Prior Art (APA) in view of Sun (Dynamic Proxy Classes).

10. **As to claim 1**, APA teaches receiving a JMI implementation request and the request associated with a metamodel (applications can access information contained in a MOF compliant model; page 5, lines 17-24 and access a metamodel; page 6, lines 20-24), the metamodel comprising at least one package (package; page 6, lines 1-18), the at least one package comprising at least one class (class; page 6, lines 1-18), the at least one class comprising at least one attribute, reference or operation (attribute, reference, method; page 6, lines 1-18). APA also teaches the user manually develops the software implementation for the JMI interfaces (page 6, lines 202-4).

11. However, APA does not teach dynamic implementing a package proxy JMI interface when the request comprises a package proxy request, dynamic implementing a class proxy JMI interface when the request comprises a class proxy request, and dynamic implementing a class instance JMI interface when the request comprises a class instance request. Sun teaches a dynamic proxy class that implements a list of interfaces specified at runtime when the class is created (page 1).

12. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of APA and Sun because it will improve the performance of

APA' system because it does not required pre-generation of the proxy class, such as with compile-time tools.

13. **As to claim 17**, APA teaches receiving a JMI implementation request and the request associated with a metamodel (applications can access information contained in a MOF compliant model; page 5, lines 17-24 and access a metamodel; page 6, lines 20-24), the metamodel comprising at least one package (package; page 6, lines 1-18), the at least one package comprising at least one class (class; page 6, lines 1-18), the at least one class comprising at least one attribute, reference or operation (attribute, reference, method; page 6, lines 1-18). APA also teaches the user manually develops the software implementation for the JMI interfaces (page 6, lines 202-4).

14. However, APA does not teach implementing a JMI interface when the JMI interface is unimplemented, and executing a stored JMI interface implementation when the JMI interface is implemented. Sun teaches implementing a list of interfaces when the list of interfaces is unimplemented (a proxy class for those interfaces will be generated dynamically and defined in the class loader; page 2), and executing a stored interface implementation when the list of interfaces is implemented (if a proxy class for the same permutation of interfaces has already been defined in the class loader, then the existing proxy class will be returned; page 2, lines 1-4 and the implementation of the dynamic proxy class API should keep a cache of generated proxy classes; page 2, last paragraph).

15. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of APA and Sun because it will improve the performance of APA' system because it does not required pre-generation of the proxy class, such as with compile-time tools and re-used the one that already implemented.

16. **As to claim 18**, APA does not teach implementing a package proxy JMI interface when the request comprises a package proxy request and when the package proxy JMI interface is unimplemented, implementing a class proxy JMI interface when the request comprises a class proxy request and when the class proxy JMI interface is unimplemented, implementing a class instance JMI interface when the request comprises a class instant request and when the class instance JMI interface is unimplemented, executing a stored packaged proxy JMI implementation when the request comprises a package proxy request and when the package proxy JMI interface is implemented, executing a stored class proxy JMI implementation when the request comprises a class proxy request and when the class proxy JMI interface is implemented, executing a stored class instance JMI implementation when the request comprises a class instance request and when the class instance JMI interface is implemented.

17. Sun teaches a dynamic proxy class that implements a list of interfaces specified at runtime when the class is created (page 1), implementing a list of interfaces when the list of interfaces is unimplemented (a proxy class for those interfaces will be generated dynamically and defined in the class loader; page 2), and executing a stored interface implementation when

the list of interfaces is implemented (if a proxy class for the same permutation of interfaces has already been defined in the class loader, then the existing proxy class will be returned; page 2, lines 1-4 and the implementation of the dynamic proxy class API should keep a cache of generated proxy classes; page 2, last paragraph).

18. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of APA and Sun because it will improve the performance of APA' system because it does not required pre-generation of the proxy class, such as with compile-time tools and re-used the one that already implemented.

19. **As to claim 34**, it is rejected under the same ground of claim 1.

20. **As to claims 50-51**, they are rejected under the same ground of claims 17 and 18.

21. Claims 2, 5, 9, 19, 22, 26, 35, 38, 42, 52, 55, and 59 are rejected under 35 U.S.C. 103(a) as being unpatentable over Admitted Prior Art (APA) in view of Sun (Dynamic Proxy Classes) further in view of Guthrie et al. (U.S. 6,385,661 B1).

22. **As to claim 2**, APA does not teach generating bytecode for a class that implements the package proxy JMI interface, creating a new instance of the class, and returning the instance. Sun teaches creating a new instance of the dynamic proxy class that implements a list of interface at run time (a proxy instance; page 1 and creating a proxy instance; page3 and proxy instance

properties; pages 3-5), and returning the instance (creating a proxy instance; page3 and proxy instance properties; pages 3-5). Guthrie teaches generating bytecode for a class that implements the interface (a reflection process ... generate the byte codes into a .class file, subject class 19; col. 5, lines 7-22). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of APA and Guthrie because it will improve the performance of APA' system by reduce development time and increases developer productivity since the developer does not have to manually generate the remote proxy classes.

23. **As to claim 5**, APA does not teach generating bytecode for a class that implements the class proxy JMI interface, creating a new instance of the class, and returning the instance. Sun teaches creating a new instance of the dynamic proxy class that implements a list of interface at run time (a proxy instance; page 1 and creating a proxy instance; page3 and proxy instance properties; pages 3-5), and returning the instance (creating a proxy instance; page3 and proxy instance properties; pages 3-5). Guthrie teaches generating bytecode for a class that implements the interface (a reflection process ... generate the byte codes into a .class file, subject class 19; col. 5, lines 7-22). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of APA and Guthrie because it will improve the performance of APA' system by reduce development time and increases developer productivity since the developer does not have to manually generate the remote proxy classes.

24. **As to claim 9**, APA does not teach generating bytecode for a class that implements the class instance JMI interface, creating a new instance of the class, and returning the instance. Sun

teaches creating a new instance of the dynamic proxy class that implements a list of interface at run time (a proxy instance; page 1 and creating a proxy instance; page3 and proxy instance properties; pages 3-5), and returning the instance (creating a proxy instance; page3 and proxy instance properties; pages 3-5). Guthrie teaches generating bytecode for a class that implements the interface (a reflection process ... generate the byte codes into a .class file, subject class 19; col. 5, lines 7-22). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of APA and Guthrie because it will improve the performance of APA' system by reduce development time and increases developer productivity since the developer does not have to manually generate the remote proxy classes.

25. **As to claims 19, 35 and 52**, see rejection of claim 2 above.

26. **As to claims 22, 38 and 55**, see rejection of claim 5 above.

27. **As to claims 26, 42 and 59**, see rejection of claim 9 above.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Diem K Cao whose telephone number is (703) 305-5220. The examiner can normally be reached on Monday - Thursday, 9:00AM - 5:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any response to this action should be mailed to:

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Diem Cao


MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100